Day: 2:

Planning the Technical foundation:

Technical Requirements:

Frontend Requirements:-

- Responsive Design: The frontend must be mobile-friendly and adapt seamlessly accross various devices (desktop, tablet, mobiles).

- User-friendly Interface :- intuitive navigation with clear product categorization search functionality, and easy checkout process.

- Performance Optimization :- fast loading times, minimal page redirects, and efficient resource management to ensure smooth user experience:-

Sanity CMS (Backend)

- Product :- Store product name, Price, description, variants, Stock and image.
- Order :- Track Order ID, Customer details, product- info, and Order Status :
- Customer Data :- Store customer name, email, Shipping address, and Order history :-

Third API's :-
integrated Api for.

1 Payment Gateway API :- integrate Third-party API's like Stripe or Paypal for secure online transactions.

2 Delivery Zone API :- use API's like Shippo or EasyPost to Calculate Shipping rates and manage delivery Zones :-

Plan for API endpoints:-

1 Product :-

Method : Get
Description : Retrieve a list of all furniture Product :-

Example :-

```
{   "Id" : 2,
    "Name" : "Wooden Dining table",
    "Category" : "Dining",
    "Price" : 250.00,
    "Description" : "A Sturdy wood dining
                     Table with 6 Chairs."
    "Material" : "wood"
    "Image" : "Url to image"
}
```

2

Customer :

Method : Post
Description : Register a new customer.

Example :-

```
{
  "Name" : "Hadiqa",
  "Email" : "abc65@gmail.com",
  "Phone" : 123456789,
  "Shipping adress : "City, House, Strelt,"
}
```

3 Checkout :

Method : Post
Description :- Process Payment and Verify the orders.

Example :-

```
{
  "Customer ID" : 2,
  "PaymentMethod" : "Credit-card",
    "CardDetails" : {
      "CardNumber" : 4131110111,
      "ExpiryDate" : 12/25,
    }
      "Order ID" : 98,
      ShippingAdress : "ABC"
}
```

Write the Technical documentation

1 System Architecture Document :-

Components :-

2 Frontend :- Provides the user interface for browsing product, manage, carts and processing order.

3 Backend :-
handles authentication Order processing and API Communication Built using node.js.

4 Data Base :- Store user data Product information, Order and transaction history :-

5 Third party API :- handle Payments, Inventory managment and Shipping Integrations :-

Workflows :-

Step 1 :- user Login/Registration Customers create account or Login.
Step 2 :- user Search and browse products.
Step 3 :- user add items to the cart and Complete Checkout.
Step 4 :- Secure Payment via third party gateways.
Step 5 :- Order is Shipping with tracking information Provided.

API Documentation:-

Product :- (Get) Retrive Product details.

Order :-(POST) Track order Status.

Checkout :-(Post) Process Payment & Finilize the Order:-

Sanity Schema :-

Product fields :-

title : String
Description : Text
Price : Number
Image : Image
Category : Reference (to product Cat

Order fields :-

user : Reference (to user)
Products : Array (of product references)
total Amount : Number
Status : String
Shipped Adress : object.

Design System Architecture.